

# 用 Node.js 滚「雪球」

孟祥宇

[mengxy@xueqiu.com](mailto:mengxy@xueqiu.com)

<http://xueqiu.com/mengxy>

# 「雪球」是什么



<http://xueqiu.com>

聪明的投资者都在这里

# 管理你的投资信息



蒙懵猛孟

- 首页
- 我的讨论
- 评论我的
- 提到我的
- 私信
- 收藏
- 关注 (692)
- 粉丝 (507)
- 自选股
- 持仓盈亏

添加自选股:

[全部](#) [指数](#) [美股](#) [A股](#) [分组](#) [自选](#) [持仓](#)

股票	当前价	涨跌额	涨跌幅	成交量	市值	目标价
LinkedIn (LNKD)	101.08	+1.55	+1.56%	191.26万	95.52亿	-
奇虎360 (QIHU)	16.50	+0.23	+1.41%	181.02万	19.69亿	-
潘多拉网络电 (P)	9.95	+0.02	+0.20%	246.93万	16.11亿	-
百度 (BIDU)	108.92	-0.38	-0.35%	426.38万	380.43亿	-
亚马逊 (AMZN)	218.37	-1.13	-0.51%	273.48万	983.80亿	¥180

[+查看更多](#)

[表情](#) [股票](#) [图片](#) [长文](#) [发布](#)

[全部](#) [讨论](#) [新闻](#) [公告](#)

# 与行业专家在线交流



赵勇研究员V:

茅台直营店“价廉物美”，短期自然受到追捧。我个人觉得茅台到明年初不会上调出厂价。茅台目前主要市场：贵州（1928吨），北京（1380吨），广东（649吨），还有华东（上海392吨，江苏468吨），河南（611吨），西南地区（四川459吨、重庆335吨）等地区。福建是188吨，销售势头还不是很强，估计氛围...

展开

中国资本市场:

关于\$贵州茅台(SH600519)\$, 对@赵勇研究员说: 请问赵先生, 我对茅台直营店比较关心, 从两个月前第一家直营店贵阳店开业当天的异常火爆, 到昨天福州店开业的相对平静, 是否预示着超高端白酒消费急剧退潮? 抑或只是地区差异? 未来三个月内是否存在提高出厂价的可能? 从长期看, 茅台的市场结构是否...

展开

06-11 12:41 相关讨论(10)

06-12 10:03 来自雪球

转发 | 收藏 | 评论

# 信息复盘



# 选择Node.js之前的思考

# 前端工程师的知识结构

HTML/CSS/JS

HTTP

# 前端工程师做什么

- 面向用户的工程师
- 浏览器/服务器不是分界线
- web/mobile



# 当时项目的状态

- 全JAVA堆栈
- 页面用jsp渲染

# Single Page App

- ajax 获取接口数据
- 用 js 拼装 json 数据



- 像 ajax 那样获取接口数据
- 像 js 那样拼装 json 数据
- 更方便的控制 html 渲染方式
- 开发顺手的工具

初试牛刀

# express <http://expressjs.com>



```
var express = require('express');
var app = express();

app.get('/', function(req, res){
  res.send('hello world');
});

app.listen(3000);
```

# node.js http module

```
http.get("http://xueqiu.com", function(res) {  
  console.log("Got response: " + res.statusCode);  
}).on('error', function(e) {  
  console.log("Got error: " + e.message);  
});
```



# {{ mustache }}

```
var view = {  
  company: "雪球",  
  positions: ["资深移动开发工程师", "资深应用开发工程师", "资深运维工程师"],  
  detail: "http://snowballfinance.com/job.html"  
}  
var output = Mustache.render("\n  
  {{company}} 在招聘: \n  
  {{#positions}}{{.}} {{/positions}} \n  
  {{{detail}}}",  
view)
```

搞定上线



经常挂掉 - -|||

# uncaughtException

```
process.on('uncaughtException', function(err) {  
  console.log('Caught exception: ' + err);  
});
```

```
        // only assign once
        if (undefined == obj[key]) {
-           obj[key] = decode(val);
+           try {
+               obj[key] = decode(val);
+           } catch (e) {
+               obj[key] = val;
+           }
        }
    });
```

<https://github.com/shtylman/node-cookie/pull/8>

还有问题...

**node 进程还在，但是不返回任何请求**

# 错误处理

```
app.get("/err", function(req, res, next) {  
  fs.readFile("file", function(err, data) {  
    if (err) {  
      throw err;  
    };  
  });  
});
```

```
app.get("/err", function(req, res, next) {  
  fs.readFile("file", function(err, data) {  
    if (err) {  
      next(err);  
    };  
  });  
});
```

**稳定了，扩大规模**

# 并发请求

```
var ep = EventProxy.create("template", "data", "l10n", function (template, data, l10n) {  
  _.template(template, data, l10n);  
});  
  
$.get("template", function (template) {  
  // something  
  ep.emit("template", template);  
});  
  
$.get("data", function (data) {  
  // something  
  ep.emit("data", data);  
});  
  
$.get("l10n", function (l10n) {  
  // something  
  ep.emit("l10n", l10n);  
});
```

# 部署工具链

- connect-assets
- uglify-js
- snockets
- sqwish
- less



# 开发调试环境

<https://github.com/undoZen/matcha>

# maxSockets

## **agent.maxSockets**

By default set to 5. Determines how many concurrent sockets the agent can have open per host.

```
// Disable socket pooling  
  
var http = require('http');  
var options = {.....};  
options.agent = false;  
var req = http.request(options)
```

全面部署

pm (<https://github.com/aleafs/pm>)

- 多进程管理器
- 基于master+worker模式
- 平滑退出和不退出前提下的进程重载

**statsd** (<https://github.com/etsy/statsd>)

# 对前端工程师的挑战

- 保证服务可靠
- 靠谱的错误处理
- 代码被并发执行带来的问题

```
app.get('/status/:id', function(req, res, next) {  
  API.get('/status.json?id=' + req.params.id, function(err, raw_status){  
    if (err) {  
      next(err);  
    }  
    status = resolve(raw_status);  
    render(status, res);  
  });  
});
```

```
app.get('/status/:id', function(req, res, next) {  
  API.get('/status.json?id=' + req.params.id, function(err, raw_status){  
    if (err) {  
      next(err);  
    }  
    var status = resolve(raw_status);  
    render(status, res);  
  });  
});
```



谢谢